THEA RENDER

physically-based unbiased, biased & gpu render engine

FLOATING LICENSING

USER MANUAL

1. INTRODUCTION

Floating licensing stands for a client-server licensing mechanism where clients can dynamically retrieve and release their license from another site, where the license server resides. This way, multiple users can have the software installed and running with the restriction that no more users/machines than the number of licenses can run the software simultaneously.
In the example of figure 1 below, there are 5 machines where the software has been installed but only 3 floating licenses. This means that no more than 3 machines can run the software simultaneously in licensed mode. In this example, Client #1, Client #2 and Client #4 have retrieved their license from the server but the other 2 clients will not run in licensed mode. Having the floating licensing gives the ability to distribute the licenses dynamically, as needed by the current user requests.

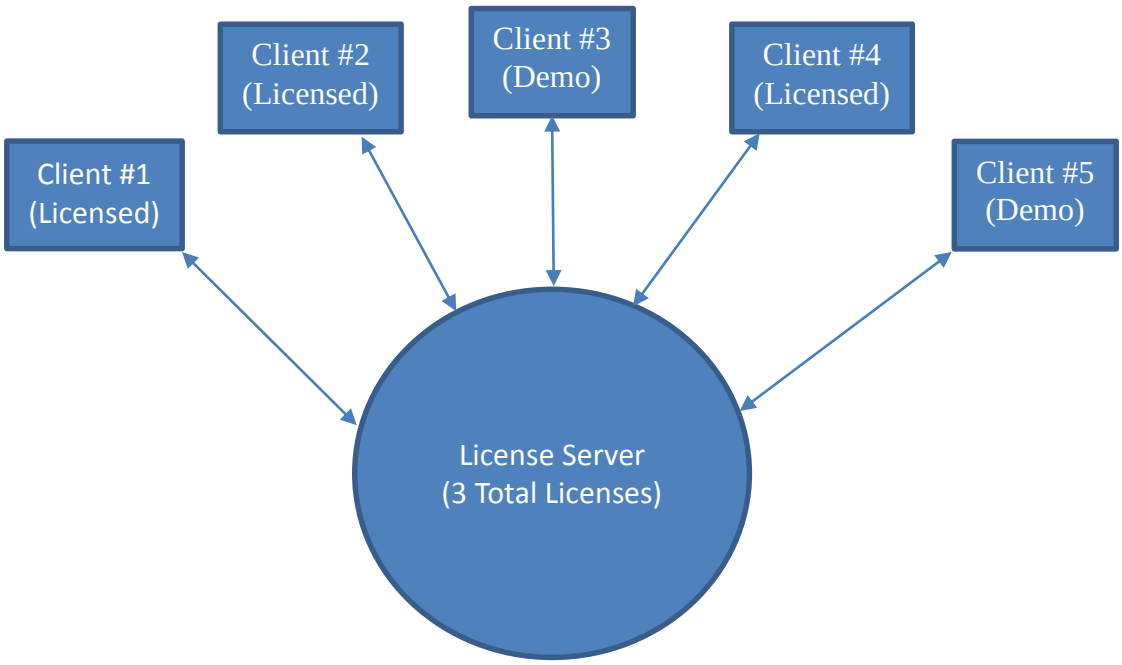

Figure 1. Example case with licensing 3 clients from the server.

For a license to be dynamically allocated to a particular client, the server must have one such floating license available in its license stock. Should all the licenses are already allocated, a new client request for license will fail and the client will run with all restrictions applied in demo mode (such as resolution limitation and watermarks). In such case, where a client must become licensed when the server has no free licenses in its stock, one of the already licensed clients must release its license to the server, in order to be subsequently acquired to the new client.

Copyright © Solid Iris Technologies

2. FLOATING LICENSING IN THEA RENDER

In Thea Render, instead of a single application, there is a complete product line that includes the standalone product (Thea Studio) along with various integrations, such as Thea for Cinema4D, Thea for SketchUp, etc. All these applications can be simultaneously running on a single machine. Communication of them directly with the license server could result in various issues of synchronization. Instead, the communication is handled by another, intermediate, application called license proxy.

In the figure below, we can see how the applications interact between 2 machines, the client – where the user runs his/her applications – and the license server.

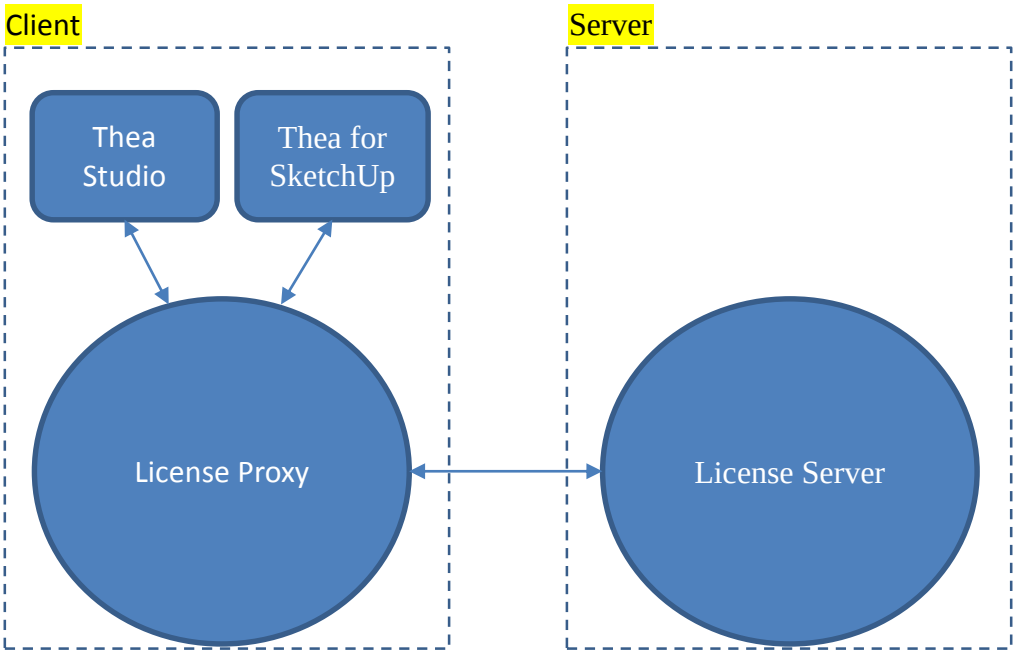

Figure 2. Typical topology of applications in the client and server machine.

With the scheme above, it is possible for the user to run multiple applications (for example, Thea Studio, Thea for SketchUp, etc.) without the need to communicate with the license server all the time. Only the first time that such a request takes place, the license proxy will communicate with the server to retrieve a license. Then, this license is reused for all Thea Render applications. Should no such applications run anymore, the proxy can release the license back to server.

The obvious benefits of the above scheme, is that there is a single point of communication between client and server which makes licensing very robust. This scheme also works in the unlucky case of an application crash. Furthermore, the proxy requires a connection to server, only at the beginning when requesting the license for the first time.

Copyright © Solid Iris Technologies

3. INSTALLATION

3.1 Windows

The License Proxy and License Server applications come as components in Thea Studio installer. In the page of choosing components, choose License Proxy (should this is an installation on a user machine), or License Server (should this is an installation on the license server machine).
After installation of these components, the License Proxy and License Server will be placed at Startup menu. This means that these applications will start automatically, immediately after the user login takes place.

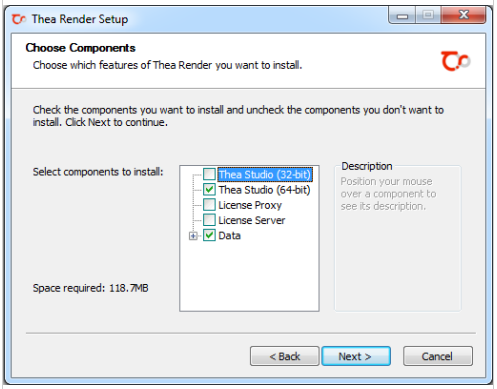
Thea Render Setup
Choose Components
Choose which features of Thea Render you want to install.
Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.
Select components to install:
Thea Studio (32-bit)
Thea Studio (64-bit)
License Proxy
License Server
Data
Description
Position your mouse over a component to see its description.
Space required: 118.7MB
< Back
Next >
Cancel

Figure 3. Choosing components

3.2 MacOSX

The License Proxy and License Server applications come as components in Thea Studio installer. These applications must start manually before any Thea Render application starts, in order to be licensed properly. Should this procedure needs to be automated, right click on either application, go to Options and in the popup menu, enable Open at Login.

Figure 4. Open at Login

3.3 Linux

The License Proxy and License Server applications come as components in Thea Studio archive. The application shortcuts will be placed on the desktop once the user installs Thea Studio. These applications must start manually before any Thea Render application starts, in order to be licensed properly.

Copyright © Solid Iris Technologies

4. Floating License Server

4.1 Using the Application

The floating license server application (titled "Thea License Server") is a small application that is typically installed on a machine that acts as a server, distributing the license on demand to client machines. Check figure 5 to see a representative screenshot of the application. In that figure, there has been already a floating license installed (but not yet allocated to a client).

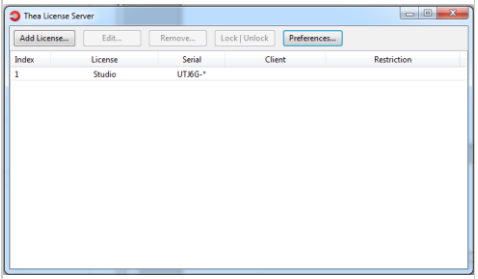

Figure 5. License server screenshot

The interface of the license server is quite simple and mainly consists of the list of installed licenses. In this list you can quickly inspect their number and type, along with the client that the license is currently assigned to.

4.2 Adding a Floating License

The very first thing you will probably want to do in the license server application is to add a floating license that you have already received. In this case, click on the Add License button. The typical Thea license form will open (figure 6a), where you need to enter your organization's name and email address along with the floating license serial. Enter the name, email and serial number and press the Activate License button. Switch to Plugins tab and enter any associated plugin licenses followed by clicking the Activate Plugin Licenses button (figure 6b).

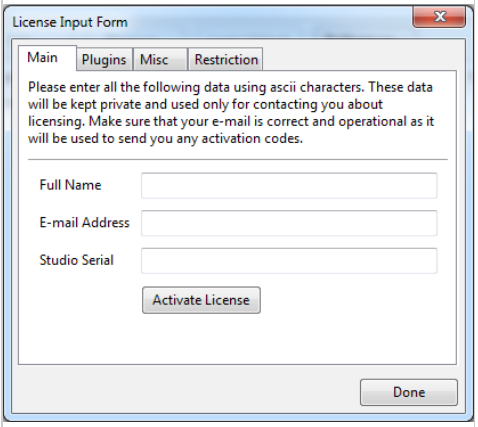
License Input Form
Main
Plugins
Misc
Restriction
Please enter all the following data using ascii characters. These data will be kept private and used only for contacting you about licensing. Make sure that your e-mail is correct and operational as it will be used to send you any activation codes.
Full Name
E-mail Address
Studio Serial
Activate License
Done

Figure 6a. Opening license form

After the license has been added and activated, click on the Done button. The license server will display now the newly added license in the list and it can use it already for any client requests.

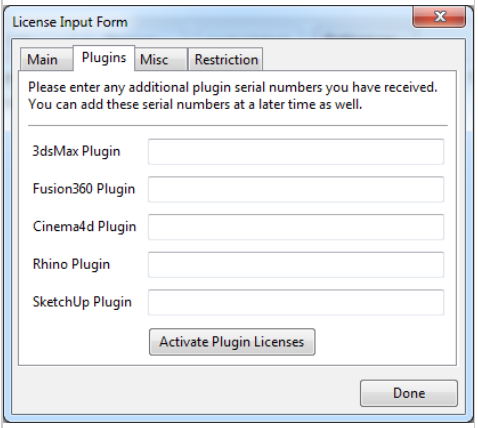

Figure 6b. Adding plugin licenses

4.3 Editing a Floating License

If you select a floating license row in the list, you will see that the Edit and Remove buttons become

Copyright © Solid Iris Technologies

enabled. Clicking then the Edit button will bring up again the license form. You can import, export or clear the license in the Misc tab. You can also assign IP addresses that will be the only ones to be accepted for becoming licensed, or, in reverse logic, IP addresses that will be rejected from becoming licensed (figure 7). This gives you a way to control each license independently, for example, by reserving a license to a particular machine or pair of machines, so as to be always available. These restrictions will appear at the last column of the license list, for quick inspection.

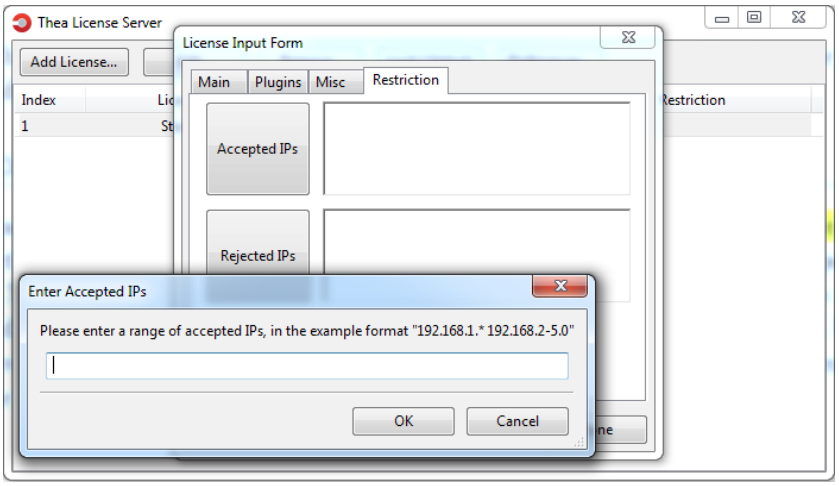

Figure 7. Screenshot of license server, during adding Accepted IPs restriction.

Besides editing a license, you can also click the Remove button that will delete the license file from the disk. Finally, you can also Lock / Unlock a license; this may be needed when you temporarily don't want to have a license assigned to anyone. Note that locking a license is not possible when the license has already been assigned to a client.

4.4 License Server Preferences

By clicking on the Preferences button, a dialog of settings pops up with a small list of parameters (figure 8). The port refers to the communication port and it must be the same in for the license server as well as the license proxy applications, communicating with the server. Note that if there is a change in the port, the license server must restart in order for the change to take effect.

There is also a global list of accepted IP addresses as well as rejected IP addresses. These can be used to block any unwanted IP addresses for all licenses (should you want to block an IP address for a specific license, you can do it from the license form Connection tab, when editing the license).

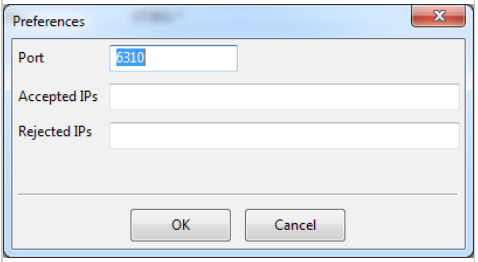

Figure 8. License server preferences

Copyright © Solid Iris Technologies

5. Floating License Proxy

5.1 Using the Application

On the client's side, all Thea applications (Studio, Plugins) communicate with the License Server through a small application called Thea License Proxy (figure 9). This application collects all requests from the applications, communicates with the license server and brings back to an application the requested license. It is important to understand that the license proxy can only cope with one license; the first time that it retrieves the license, it reuses that one to license any subsequent application requests.

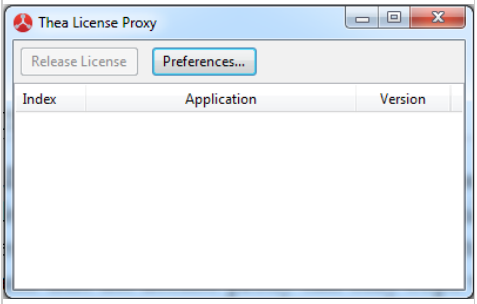

Figure 9. License proxy screenshot

If all applications using a license are closed then the license will not be needed anymore. The license proxy will, by default, auto release this license back to server after 10 seconds. The user may also release the license immediately, after pressing the Release License button.

5.2 License Proxy Preferences

By clicking the Preferences button, a new dialog pops up with various parameters for the license proxy (figure 10).
The Server Port number is the communication port with the license server. Note that the next port number (i.e. 6311 in this case) is also being used for communication with the applications. Obviously, these two ports should not be in use by any other network application. If the port changes, both license proxy and all Thea applications need to restart for the new port to take effect.
The Server Address is the IP address where the server is located. Since this is a fixed IP address, the server must be located at a machine that its IP is fixed and not dynamically allocated.
The Auto Release, along with the Release Timeout, controls the auto release of a license, after all Thea applications have been closed. You may disable Auto Release if you want to have the license reserved for more time, until for example you restart a Thea application.

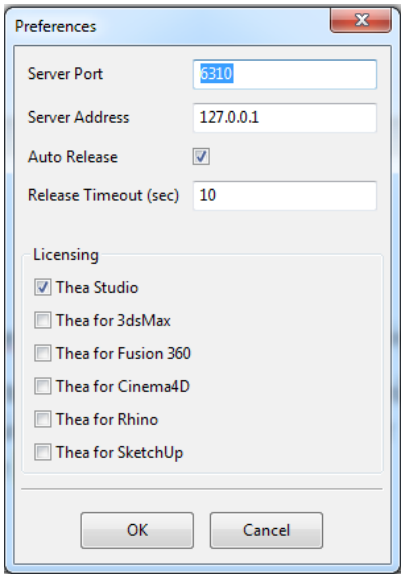

Figure 10. License proxy preferences

Finally, the Licensing group has all Thea applications that the requested license should cover. You should check all the applications that may be used at any time (and you know that they can be licensed). For example, if your floating licenses are for “Thea for SketchUp”, then you should check both Studio and SketchUp. A requested license will then include both Studio and plugin, even in the case you run Studio; this way, the license allocated can also be used for the plugin, launched at a later time.

Copyright © Solid Iris Technologies